Importance of Dataspace Embeddings when Evaluating Text Clustering Methods

Alain Lelu, Martine Cadot

Importance of Dataspace Embeddings when Evaluating Text Clustering Methods

Alain LELU and Martine CADOT

Abstract Fair evaluation of text clustering methods needs to clarify the relations between 1)pre-processing, resulting in raw term occurrence vectors, 2)data transformation, and 3)method in the strict sense. We have tried to empirically compare a dozen well-known methods and variants in a protocol crossing three contrasted open-access corpora in a few tens transformed dataspaces. We compared the resulting clusterings to their supposed "ground-truth" classes by means of four usual indices. The results show both a confirmation of well-established implicit combinations, and good performances of unexpected ones, mostly in spectral or kernel dataspaces. The rich material resulting from these some 600 runs includes a wealth of intriguing facts, which needs further research on the specificities of text corpora in relation to methods and dataspaces.

Key words: evaluation method, method comparison, text clustering, K-Means, Normalized Matrix Factorization, Latent Dirichlet Allocation , hierarchical clustering, linkage method, spectral clustering, graph partition, kernel clustering, Salton tf-idf, Okapi tf-idf, chi-square metrics, Laplacian spectral decomposition, correspondence analysis, kernel expansion

1 Introduction : motivations and goals

Evaluation of text clustering methods is one of the key issues in the problem of bibliometric delineation of scientific fields. As co-authors of [1] we have tried to test seventeen clustering methods with a publicly available real-life test set, the

Alain LELU
Université de Franche-Comté (rtd), e-mail: alelu@orange.fr

Martine CADOT
LORIA Nancy France, e-mail: martine.cadot@loria.fr

Reuters' test bench [2] which adds up several difficulties of text clustering, i.e. mainly strongly unbalanced man-made classes (the targeted "ground-truth"), and texts of unbalanced sizes. An unexpected result was that antique agglomerative methods, especially Ward hierarchical clustering, performed better than many more recent ones. Was it the case for all types of corpora? Above all we realized that for the sake of fair comparisons, as well as conceptual clarity, we should clearly separate the transformations of the raw word-count data (for example into Salton tf-idf vector representation, or Laplacian spectral space, etc.) from the algorithms in the strict sense, instead of using long-time accepted implicit combinations. For example, no rationale forbid using Non-negative Matrix Factorization in a spectral space. This consideration is in line with the conceptual clarifications operated in [3]; last, but not least, unexpected recommendations may proceed from non-classic combinations. This clarification is one of our guiding threads, and led us to the study and report [4] we submitted to the Neutral Cluster Benchmarking Challenge, organized by the Cluster Benchmarking Task Force of the IFCS, which won the Challenge.

Though restricting our scope to text clustering, it is clear that many types of texts need now to be processed: abstracts or plain texts of scientific papers, which are our primary scientific interest, or journal, litterary or legal texts, or texts originating in the social nature of Internet communications, such as contributions to forum discussions, or social networks. We decided to base our present survey on three typical and contrasted test sets: a full-text scientific database, a wire of press agency, and an Internet discussion forum. It is clear that the complete text preprocessing chain is out of research goal, so we have to rest on one same linguistic – or weakly linguistic – term, lemma or stem extraction scheme, and same elimination of infrequent or too frequent words. This point must not keep us from exploring the influence of truncating the resulting vocabulary in chosen distribution quantiles, contrary to usual benchmark studies which merely mention an absolute occurrence threshold. All these specifications led us to the choices we expose in the methodology section.

Of course the options on methods and types of dataspaces to be considered are inevitably somehow arbitrary: we tried to take account of the most usual algorithms, or method families, such as K-means, hierarchical agglomerative clustering, spectral clustering, graph clustering, kernel clustering, and we added two more specific methods, i.e. Non-negative Matrix Factorization (NMF) and Latent Dirichlet Allocation (LDA), which amounts to a dozen methods and variants.

Concerning dataspaces, we chose to add to the plain term occurrence vector space the transformed spaces by Salton's and Okapi's tf-idf weighting schemes, by chi-square metrics, by Laplacian spectral decomposition, by Correspondence Analysis factors, and last by order-2 polynomial kernel expansion.

Given the combinatorics of the three main elements – text types, dataspaces and algorithms – our research study could be nothing but an exploration, strongly constrained by available resources. However, some interesting conclusions will be drawn out this exploration. In the final conclusion, we will deal with what may be continued and deepened in our perspective, given the results.

Let us close this introduction saying that we are indebted to the remarkable initiative of the Brazilian LABIC team [5] who homogeneously pre-processed [6] some

forty text collections and made the documents-by-terms matrices available online on their site [7].

2 Methodology

To evaluate non supervised classifications via a methodology devoted to supervised ones is an imposed solution, for want of anything better. It may at least result, in default of a universal ranking of methods, in fruitful reflections on the typology of texts, or the nature of the human categorization and abstraction process and its similarities and differences with methods mostly optimizing an intrinsic objective function. Another core imperative we have set is transparency and reproducibility: in addition to the direct link to the documents-by-terms matrices we have provided above, the complementary material in HAL site [4] gathers the links to the public-access code we used. Though most algorithms are theoretically insensitive to the ordering of input vectors, in practice we experienced that tied effects, among others, could affect the results. This is why we have randomly scrambled the data vectors.

Choice of test corpora

The three ”prototype” test corpora mentioned in the introduction are, first, Reuters’ ”ModApté Split” [8] limited to the eight most important classes (”Re8” in the present study, 7674 documents, 8901 terms), second, the ACM collection made of the proceedings of forty conferences in different computer science areas (3493 papers, 60 768 terms), third, the ”20 Newsgroups” collection (”Ng20”) composed of 18 808 messages posted in twenty Usenet groups (45 434 terms). The size of the man-made reference classes is strongly unbalanced in the case of Re8 (two of them constitute 81% of the documents), roughly equal in the case of ACM and Ng20. It is to be noted that the sole Reuters’ class labels are issued from a direct manual indexing. The two others origin in the concatenation of sub-corpora of comparable size. They could therefore be considered ”semi-real-world data”, not really representative of real-life non-annotated corpora

Truncating the vocabularies

As the size of the vocabularies are unbalanced (Re8: about 8900 terms, ACM: 60 800 terms, Ng20: 45 000 terms) but extensive (the hapaxes, i.e. terms of total occurrence one, are included in this count), we decided a common scheme for a vocabulary-independent truncation by thresholds: in addition to the basic option of retaining the whole vocabulary, we built two sub-corpora per test corpus retaining the third quartile of the term distribution (25% of the total occurrences), and the seventh ”octile” (12.5%).

Choice of clustering methods

For the bibliographic references to the methods, see [4]. We have affected a lower priority to algorithms with two parameters (DbScan, Affinity Propagation, Smart Local Moving Algorithm), or one parameter with deceptive results on Reuters' corpus (Density Peaks, Independent Component Analysis, Fuzzy c-means, K-Means++). We selected:

Plain K-means clustering ("KM"). We implemented 20 elementary runs, or "replicates", per run, selecting the best one in terms of the local optimum of the K-means intrinsic objective function.

Hierarchical agglomerative clustering with two linkage variants: average link ("HCa"), and Ward ("HCw"). Originally in O(#documents^3) time complexity, more recent contributions have lowered this constraint to O(#documents^2) [9].

Spectral clustering. We used the "standard" combination K-means/Laplacian spectral dataspace, but also explored (with success, see below) many other combinations.

Graph clustering methods. We chose the two most broadly recognized ones, i.e. Louvain and InfoMap. Note that these methods, in contrast to all the tested other ones, do not need fixing a required number of clusters, hence a major operational advantage when no idea of the "true number of clusters" is known beforehand - hierarchical clustering being in an intermediate position, as in one run it leaves the choice of the cluster number to the user

Non-negative Matrix Factorization ("NMF"). This decomposition is akin to be used as a clustering method, when the label of a document is attributed as the axis number of its maximum projection. As this method converges to local optima of its objective function, we implemented the same "20 runs" strategy as for K-means.

Latent Dirichlet Allocation ("LDA") is well-known and much respected as deeply founded in theoretic grounds.

Kernel clustering. Thanks to the "kernel trick", a documents-by-documents similarity matrix ("Gram matrix") is built without explicit expansion of the raw dataspace by a kernel function. Here we used an order-2 polynomial kernel, which amounts to take into account the wholeness of the 2-term itemsets in each document when comparing one to another. In this case the raw dataspace is not made of numeric occurrence vectors, but of binary existence ones.

Choice of dataspaces

For the mathematical formulations, see [4]. In addition to the plain term-occurrence vector space, we have considered and built:

- Salton's vector space, weighted by the classic tf-idf scheme
- Okapi (also coined BM25) vector space, with a more cryptic, but statistically grounded, weighting scheme [10]
- Chi-square metrics, which amounts to a Euclidean vector space with transformed vectors as specified in [11]
- Laplacian spectral space [12]

- Correspondence Analysis spectral space (”CA space”) [13]
- Kernel space : Given the much contrasted values in the Gram similarity matrix, the cosine distance is well-fit to this dataspace [14]

Note that Euclidean distances in the complete CA factor space equal chi-square distances [13]. Therefore, truncating this space by considering the sole most informative factors amounts to consider ”partial chi-square” distances, a priori more relevant than chi-square distances. These six transformations of a documents-by-terms matrix are convenient for the KM, NMF, LDA and Spectral Clustering methods. Other methods, such as Hierarchical Clustering, Graph methods and Kernel methods, need a documents-by-terms similarity (or dissimilarity) matrix. Depending on each dataspace-method combination, we have used Euclidean distance or “cosine” distance (i.e. 1-cosine, which weights half of the squared chord distance).

Choice of evaluation measures

We chose the four most usual indices encountered in the evaluation literature, i.e. first, Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI), which compute independently from the number and labels of clusters; and second, mean local class-vs.-cluster F-scores (F) and global Purity score (i.e. 1-global error rate) which need the same number of clusters and classes, and same labels. We have aligned the k classes and the k most ”analogue” clusters, in the sense of local F-scores, by means of the ranking issued from the leading factor of the Correspondence Analysis of the classes-by-clusters F-score matrix.

Code implementation and computer efficiency

As computer efficiency is out of our goals, we implemented the data transformations, method code, and post-processing code in an Octave environment, on an Intel 6-core I7, 3.33GHz, 48Go RAM computer. Method codes were derived from existing Matlab® codes (links to the original pieces of code are available in the supplementary material). Their degree of computing time optimization varies considerably: e.g. in the case of the 19 000 documents Ng20 test set, from 2 minutes for twenty elementary runs of the standard ”litekmeans.m” code (itself implementing 20 ”replicates”), to 6 hours for one run of Louvain method...and 24h for one run of Hierarchical average-link clustering.

3 Evaluation results

We have tried as much as possible to cross-combine corpora $\times$ data transformations $\times$ methods. This was not always possible, due to constraints such as computing time or resources devoted to systematically poor results. Our reference site [4] displays the entirety of the results in 29 figures. Fig.1 is one example.

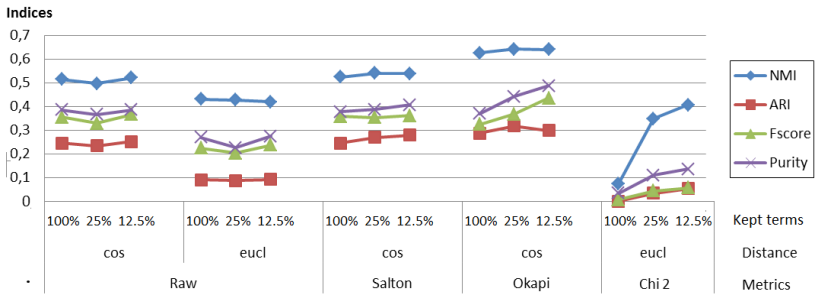

Fig. 1 K-means on ACM corpus

Let us focus now on the measurement tools: we can observe that in the case of the two "balanced" corpora, the four evaluation indices behave in a much parallel and orderly manner. In contrast, this parallelism and regular ranking deteriorate in the Reuters'8 unbalanced corpus, and to a lesser extent when hierarchical methods are used. A thorough investigation could perhaps explain these interesting discrepancies, but is clearly out of our present goals. We have thus chosen the most stable NMI index as a reference measure for ranking each corpus' runs (ACM: 246 runs, Re8: 237 runs, Ng20: 109 runs, summing up to 592 runs).

	ACM			*Reuters'8*			*20 NewsGroups*		
	k=40, 3493 docs, 60768 terms			*k=8, 7674 docs, 8901 terms*			*k=20, 18808 docs, 45434 terms*		
Top methods	**1)HC-Ward**	**2)HC-Ward**	**3)K-Means**	**1)K-Means**	**2)HC-ave.**	**3)HC-Ward**	**1)NMF**	**2)HC-Ward**	**3)K-Means**
Dataspace type	Spectral Lapl.	Standard	Spect. Lapl.	Standard	Spect. CA	Kernel	Std	Spectral CA	Standard
#factor or order	40; 80 (=k; 2k)	40 (=k)	80 (=2k)	8 (=k)	8 (=k)	Polynom. 2	20 (=k)	40 (=2k)	20 (=k)
% vocabulary	100%; 12.5%	100.%	12.5%	100%; 12.5%	12.5%	12.5%	100%	100%	12.5-100-25
Metrics	Salton	Okapi	Okapi	Okapi; Salton	Raw	Raw	Okapi	Raw	Salton
NMI	.6980-.6714	.6739	.6712	.6461-.6314	.629	.625	.6252	.6220	.621
Comput. time	77s	75s	6s	10s	24h	800s	107s	4h	150s

Fig. 2 "Top three" methods (NMI criterion) for each corpus

The best runs of Fig.2 clearly depend on the corpora. A large variety of dataspace transformations (truncated or not vocabulary, Salton's, 0kapi, or raw dataspace, kernel or Laplacian spectral space, ...) and methods (HC-Ward, K-Means, NMF) are present. It can be noted that four methods only upon nine may be considered "classic", i.e. K-Means on Salton's or Okapi dataspace, standard Hierarchical Clustering and Non-negative Matrix Factorization.

Examining the "top 50" runs of each corpus (ranked by decreasing NMI values), a few common behaviors emerge:

Partial commonalities:

Concerning ACM corpus: the spectral HC-Ward method in all the variants of the Laplacian Okapi-weighted space dominates massively; the spectral KMeans in the

same spaces appears in the top-50 list nine times, standard HC Ward and standard NMF appear two times and three times respectively.

Concerning Re8 corpus: Okapi-weighted or Salton-weighted standard K-Means dominate, followed by spectral HC-average in the CA factor space, then by spectral K-Means in the CA (sometimes Laplacian) factor space; three kernel HC-Ward appear in the list, as well as six standard NMF and three Louvain methods in the Salton space.

Concerning Ng20 corpus: standard K-Means comes out on top, together with NMF Okapi in the CA factor space and spectral HC Ward in the same space. Next come Salton-weighted (sometimes Okapi-weighted) spectral K-Means, and spectral HC Ward in CA space. Kernel HC Ward ranks last.

Global commonalities

As far as inter-corpora comparisons are concerned, we constructed a relative performance indicator by dividing the NMI of a given "data space + method" combination by the maximum NMI observed on this corpus. With three values per combination, we can provide a heuristic view of the overall performance of a particular combination by calculating the average and the maximum range for those three values. The following lines summarize this process for the three combinations which to our view achieve the best compromise between performance and independence from the corpus (embodied in low range values):

1. Standard NMF on Okapi weighted data, with non-truncated vocabulary: relative NMI: 95.4%, range: 7.9%
2. Spectral hierarchical clustering (with Ward link) in the space of the $2k$ leading CA factors (k is the number of required clusters), with a strong truncation of the vocabulary (12.5% of the original vocabulary): relative NMI: 92.0%, range: 9.1%
3. Standard K-Means on Okapi weighted data, and vocabulary also truncated at 12.5%: relative NMI: 91.1%, range: 18.2%

The main problem for one to follow recommendation 2 is to build the spectral space for big real-life data. In many computer languages indeed, efficient sparse Singular Value Decomposition procedures exist, appropriate when the problem is to draw a limited number of main eigenvalues and eigenvectors from huge datatables, which is the case in the present study. Otherwise parallel graphics co-processors may be dedicated to this task.

4 Conclusions and perspectives

We hope we have brought some clarification to the problem of evaluating text clustering procedures, by considering separately the algorithms and the dataspaces in which they operate. We have achieved some 600 runs of a dozen algorithms and

variants, in a few tens various dataspaces, on three prototypical and public access test corpora. We have brought to light an unexpected variety of optimal combinations of methods and dataspaces, from which we have derived three cautious recommendations. The variety of possible transformations and parameters requires a considerable continuation effort for improving our understanding and mastery of artificial vs. human categorization processes. We hope that this empirical survey will contribute to such an issue. In a modest first step, we will explore the influence of linguistic pre-processing: choice or elimination of word categories, comparison between taking into account multi-word expressions and kernel expansion of uniterms.

References

1. M. Zitt, A. Lelu, M. Cadot, and G. Cabanac, "Bibliometric delineation of scientific fields," in *Handbook of Science and Technology Indicators*, ser. Handbook of Science and Technology Indicators, W. Glänzel, H. F. Moed, U. Schmoch, and M. Thelwall, Eds. Springer International Publishing, 2018. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01942528
2. D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "Rcv1: A new benchmark collection for text categorization research," *Journal of machine learning research*, vol. 5, no. Apr, pp. 361–397, 2004. [Online]. Available: http://www.daviddlewis.com/resources/testcollections/rcv1/
3. I. Van Mechelen, A.-L. Boulesteix, R. Dangl, N. Dean, I. Guyon, C. Hennig, F. Leisch, and D. Steinley, "Benchmarking in cluster analysis: A white paper," *arXiv preprint arXiv:1809.10496*, 2018.
4. A. Lelu and M. Cadot, "Evaluation of text clustering methods and their dataspace embeddings: an exploration," in *IFCS 2019 - 16th International of the Federation of Classification Societies*, Thessaloniki, Greece, Aug. 2019. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02116493
5. R. G. Rossi, R. M. Marcacini, S. O. Rezende *et al.*, "Benchmarking text collections for classification and clustering tasks." 2013.
6. LABIC, checked on January 30, 2020. [Online]. Available: http://sites.labic.icmc.usp.br/tpt/
7. LABIC, checked on January 30, 2020. [Online]. Available: http://sites.labic.icmc.usp.br/text_collections/
8. C. Apté, F. Damerau, and S. M. Weiss, "Automated learning of decision rules for text categorization," *ACM Trans. Inf. Syst.*, vol. 12, no. 3, p. 233–251, Jul. 1994. [Online]. Available: https://doi.org/10.1145/183422.183423
9. F. Murtagh, "Complexities of hierarchic clustering algorithms: state of the art," *Computational Statistics Quarterly*, vol. 1, no. 2, pp. 101–113, 1984.
10. S. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford, "Okapi at trec-2," *NIST SPECIAL PUBLICATION SP*, pp. 21–21, 1994.
11. P. Legendre and E. D. Gallagher, "Ecologically meaningful transformations for ordination of species data," *Oecologia*, vol. 129, no. 2, pp. 271–280, 2001.
12. U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.
13. J. Benzécri, "L'analyse des correspondances. l'analyse des données," *Dunod. Paris*, vol. 2, 1973.
14. M. Girolami, "Mercer kernel-based clustering in feature space," *IEEE Transactions on Neural Networks*, vol. 13, no. 3, pp. 780–784, 2002.